

# Compte Rendu du TP9

## Encadré par Akssase

### Réalisé par Youssef Oubrik - Ginfo4

December 15, 2024

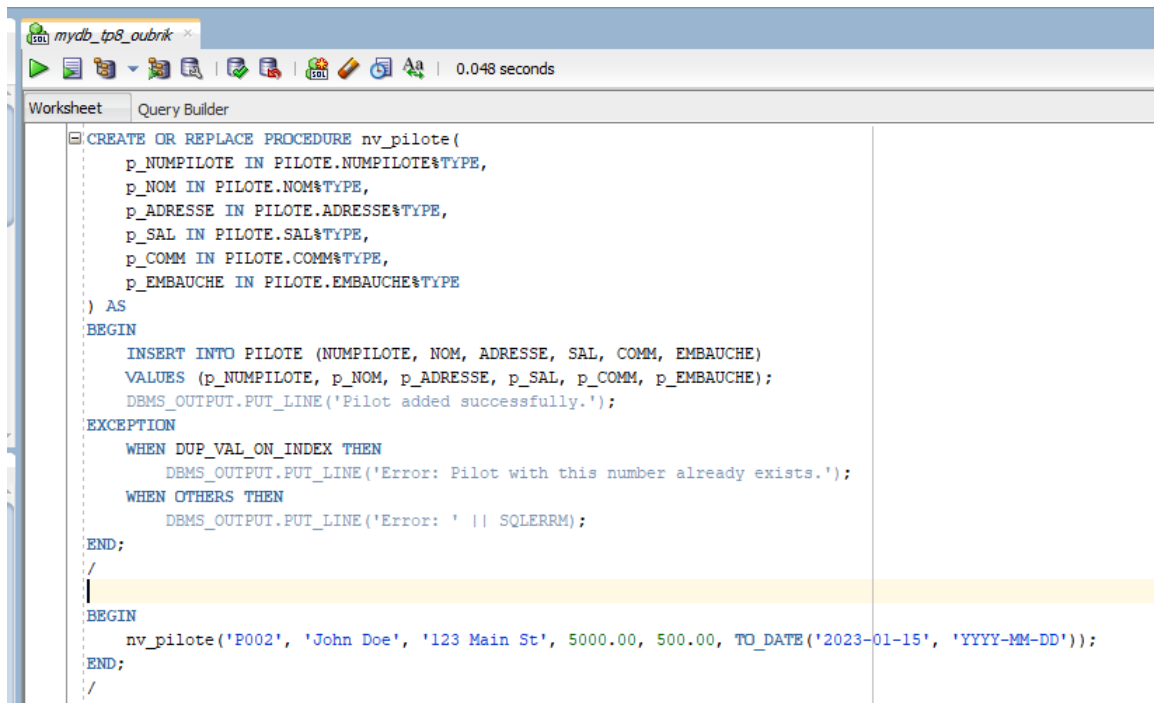
## Objectifs du TP9

Le TP9 vise à approfondir les concepts de gestion des bases de données relationnelles avec des exercices pratiques sur :

- Création et gestion de procédures stockées.
- Développement de fonctions stockées.
- Implémentation de déclencheurs.
- Utilisation de packages et transactions.

# Procédures Stockées

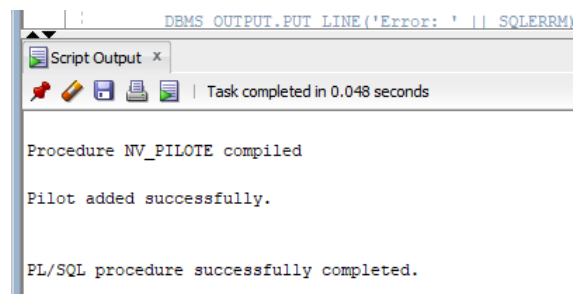
## 1. nv\_pilote :



```
CREATE OR REPLACE PROCEDURE nv_pilote(  
  p_NUMPILOTE IN PILOTE.NUMPILOTE%TYPE,  
  p_NOM IN PILOTE.NOM%TYPE,  
  p_ADRESSE IN PILOTE.ADRESSE%TYPE,  
  p_SAL IN PILOTE.SAL%TYPE,  
  p_COMM IN PILOTE.COMM%TYPE,  
  p_EMBAUCHE IN PILOTE.EMBAUCHE%TYPE  
) AS  
BEGIN  
  INSERT INTO PILOTE (NUMPILOTE, NOM, ADRESSE, SAL, COMM, EMBAUCHE)  
  VALUES (p_NUMPILOTE, p_NOM, p_ADRESSE, p_SAL, p_COMM, p_EMBAUCHE);  
  DBMS_OUTPUT.PUT_LINE('Pilot added successfully.');
```

```
EXCEPTION  
  WHEN DUP_VAL_ON_INDEX THEN  
    DBMS_OUTPUT.PUT_LINE('Error: Pilot with this number already exists.');
```

```
  WHEN OTHERS THEN  
    DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);  
END;  
/  
  
BEGIN  
  nv_pilote('P002', 'John Doe', '123 Main St', 5000.00, 500.00, TO_DATE('2023-01-15', 'YYYY-MM-DD'));  
END;  
/
```

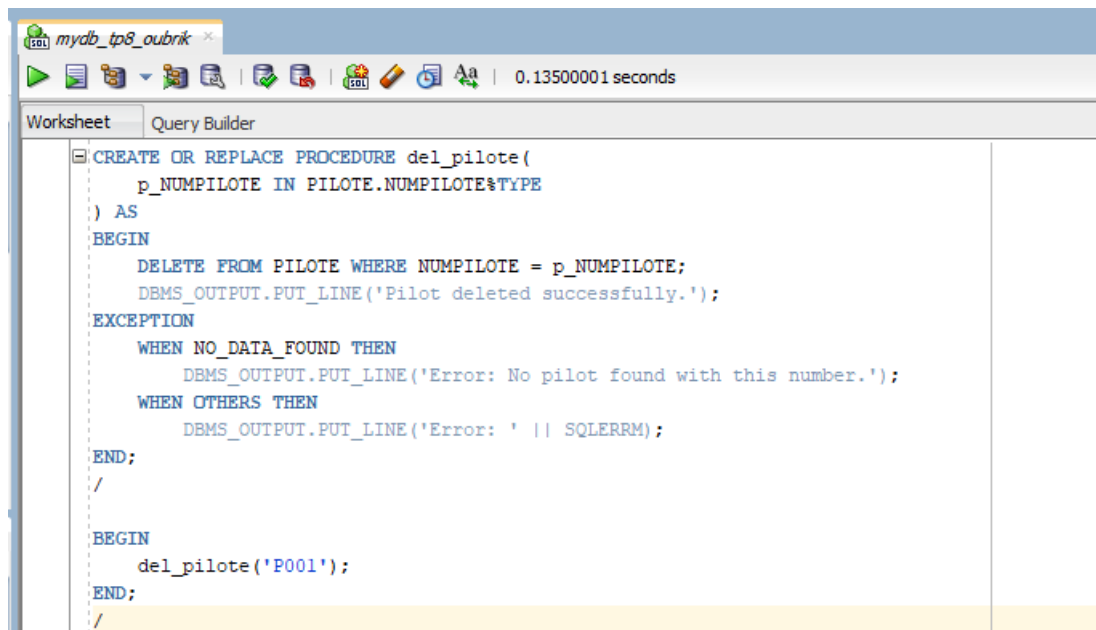


```
DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM)
```

```
Script Output x  
Task completed in 0.048 seconds
```

```
Procedure NV_PILOTE compiled  
Pilot added successfully.  
PL/SQL procedure successfully completed.
```

## 2. del\_pilote :



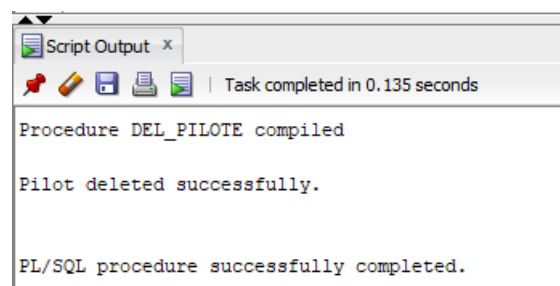
The screenshot shows a SQL IDE window titled 'mydb\_tp8\_oubrik'. The 'Query Builder' tab is active, displaying the following PL/SQL code:

```
CREATE OR REPLACE PROCEDURE del_pilote(  
  p_NUMPILOTE IN PILOTE.NUMPILOTE%TYPE  
) AS  
BEGIN  
  DELETE FROM PILOTE WHERE NUMPILOTE = p_NUMPILOTE;  
  DBMS_OUTPUT.PUT_LINE('Pilot deleted successfully.');
```

The code continues with an exception block and a call to the procedure:

```
EXCEPTION  
  WHEN NO_DATA_FOUND THEN  
    DBMS_OUTPUT.PUT_LINE('Error: No pilot found with this number.');
```

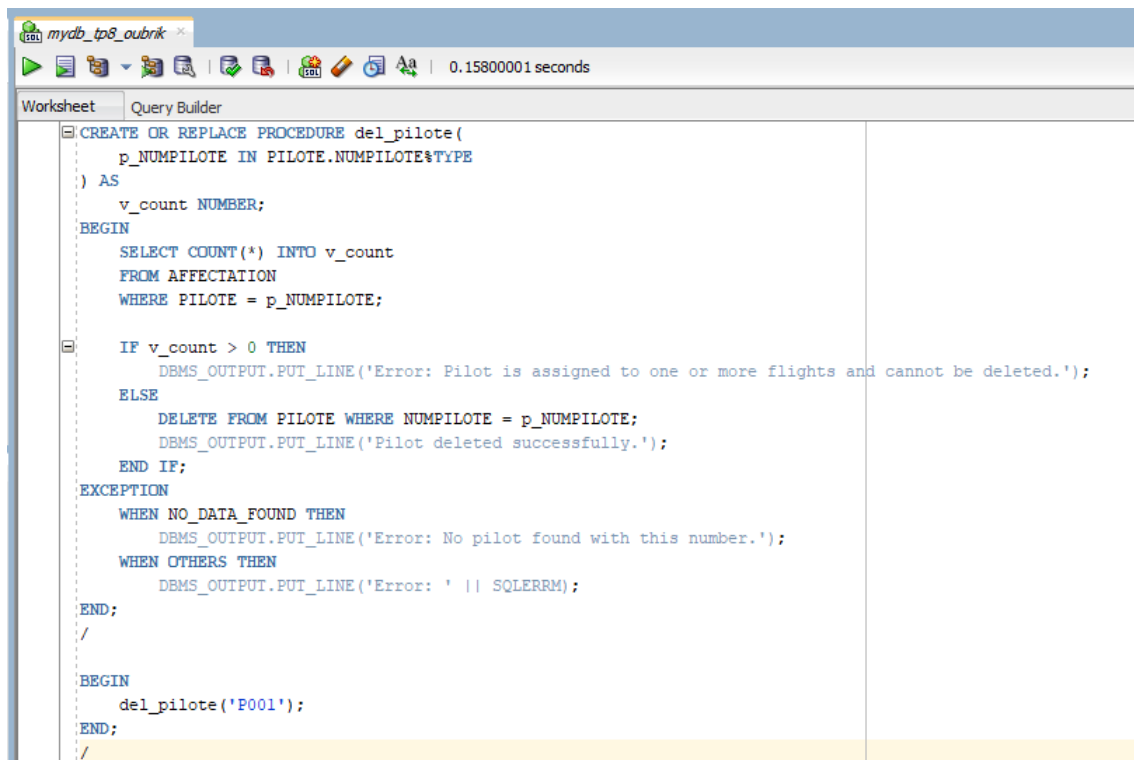
```
  WHEN OTHERS THEN  
    DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);  
END;  
/  
  
BEGIN  
  del_pilote('P001');  
END;  
/
```



The 'Script Output' window shows the following output:

```
Procedure DEL_PILOTE compiled  
  
Pilot deleted successfully.  
  
PL/SQL procedure successfully completed.
```

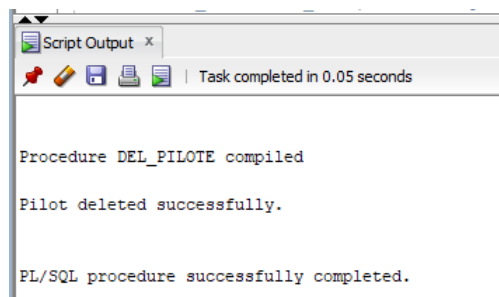
### 3. Modification de del\_pilote :



The screenshot shows the Oracle SQL Developer interface. The main window is titled 'mydb\_tp8\_oubrik' and displays a PL/SQL procedure named 'del\_pilote'. The procedure is defined as follows:

```
CREATE OR REPLACE PROCEDURE del_pilote(  
  p_NUMPILOTE IN PILOTE.NUMPILOTE%TYPE  
) AS  
  v_count NUMBER;  
BEGIN  
  SELECT COUNT(*) INTO v_count  
  FROM AFFECTATION  
  WHERE PILOTE = p_NUMPILOTE;  
  
  IF v_count > 0 THEN  
    DBMS_OUTPUT.PUT_LINE('Error: Pilot is assigned to one or more flights and cannot be deleted.');  ELSE  
    DELETE FROM PILOTE WHERE NUMPILOTE = p_NUMPILOTE;  
    DBMS_OUTPUT.PUT_LINE('Pilot deleted successfully.');  END IF;  
EXCEPTION  
  WHEN NO_DATA_FOUND THEN  
    DBMS_OUTPUT.PUT_LINE('Error: No pilot found with this number.');  WHEN OTHERS THEN  
    DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);  
END;  
/  
  
BEGIN  
  del_pilote('P001');  
END;  
/
```

The procedure is designed to delete a pilot from the PILOTE table based on their number. It includes error handling for cases where the pilot is assigned to flights or does not exist.



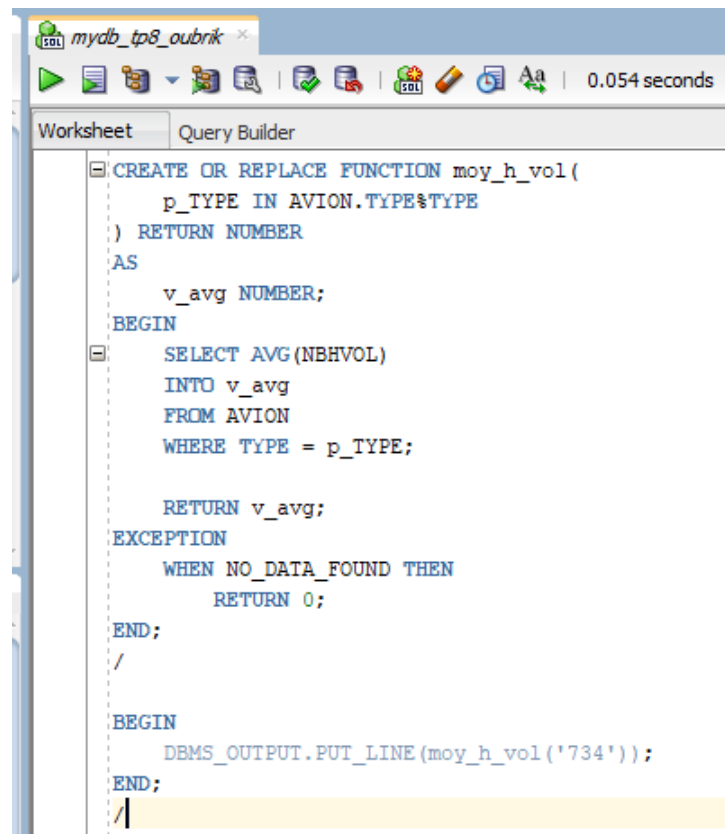
The screenshot shows the 'Script Output' window, which displays the results of the PL/SQL procedure execution. The output is as follows:

```
Procedure DEL_PILOTE compiled  
  
Pilot deleted successfully.  
  
PL/SQL procedure successfully completed.
```

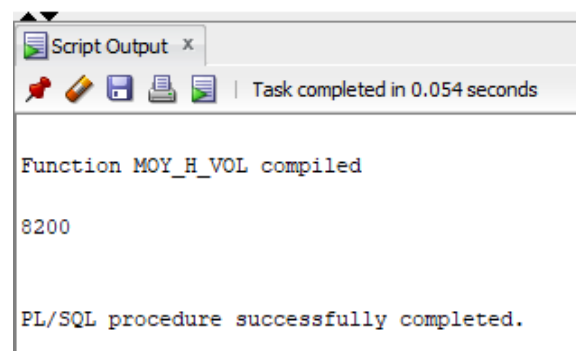
The output confirms that the procedure was compiled successfully and that the pilot with the number 'P001' was deleted successfully.

## Fonctions Stockées

### 4. fonction moy\_h\_vol :



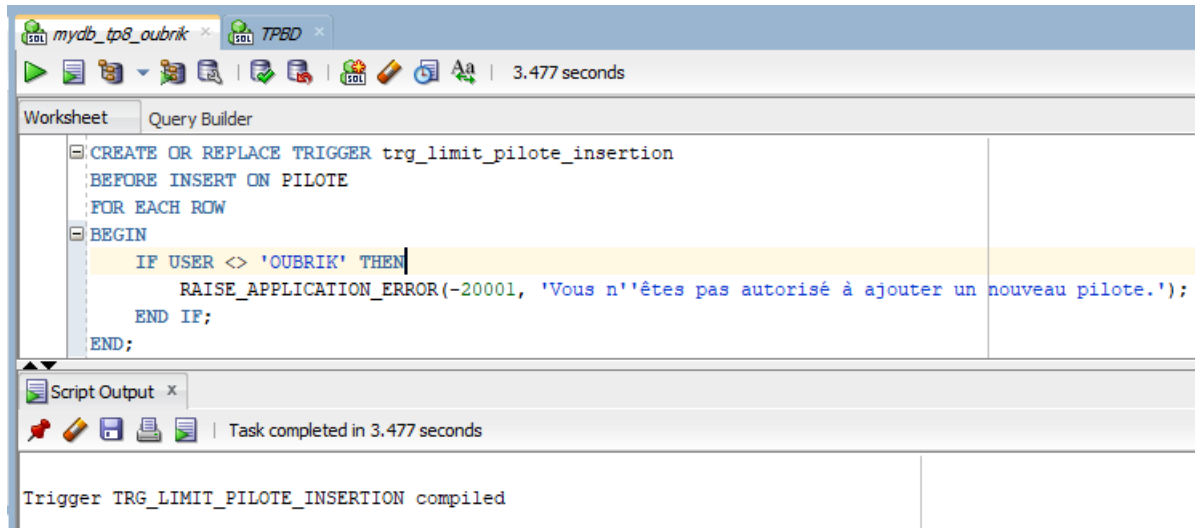
```
CREATE OR REPLACE FUNCTION moy_h_vol(  
  p_TYPE IN AVION.TYPE%TYPE  
) RETURN NUMBER  
AS  
  v_avg NUMBER;  
BEGIN  
  SELECT AVG(NBHVOL)  
  INTO v_avg  
  FROM AVION  
  WHERE TYPE = p_TYPE;  
  
  RETURN v_avg;  
EXCEPTION  
  WHEN NO_DATA_FOUND THEN  
    RETURN 0;  
END;  
/  
  
BEGIN  
  DBMS_OUTPUT.PUT_LINE(moy_h_vol('734'));  
END;  
/
```



```
Script Output x  
Task completed in 0.054 seconds  
  
Function MOY_H_VOL compiled  
  
8200  
  
PL/SQL procedure successfully completed.
```

# Triggers

## 5. Limitation de l'ajout d'un nouveau pilote :

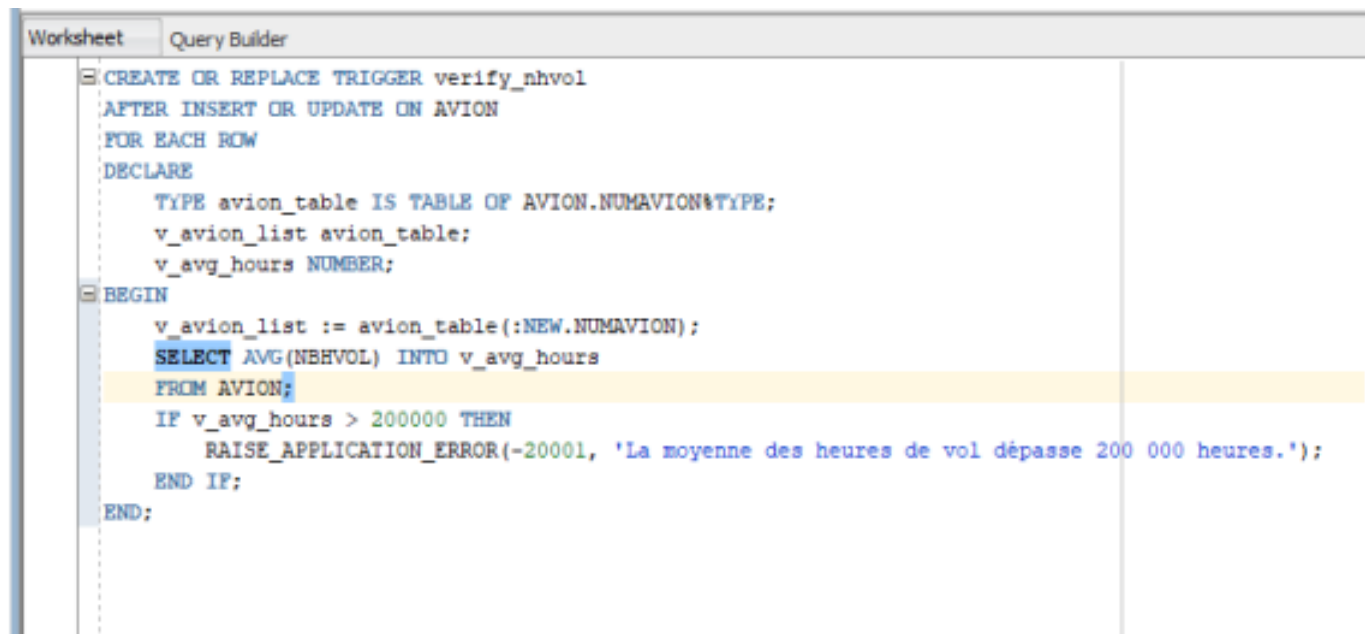


The screenshot shows the SQL Developer interface with a query window titled 'Query Builder'. The query is for creating a trigger named 'trg\_limit\_pilote\_insertion'. The trigger is set to fire 'BEFORE INSERT ON PILOTE' and is configured to fire 'FOR EACH ROW'. The trigger body contains an 'IF' statement that checks if the user is not 'OUBRIK'. If true, it raises an application error with the message 'Vous n'êtes pas autorisé à ajouter un nouveau pilote.'.

```
CREATE OR REPLACE TRIGGER trg_limit_pilote_insertion
BEFORE INSERT ON PILOTE
FOR EACH ROW
BEGIN
    IF USER <> 'OUBRIK' THEN
        RAISE_APPLICATION_ERROR(-20001, 'Vous n'êtes pas autorisé à ajouter un nouveau pilote.');
```

Below the query window, the 'Script Output' window shows the message: 'Trigger TRG\_LIMIT\_PILOTE\_INSERTION compiled'. The task completed in 3.477 seconds.

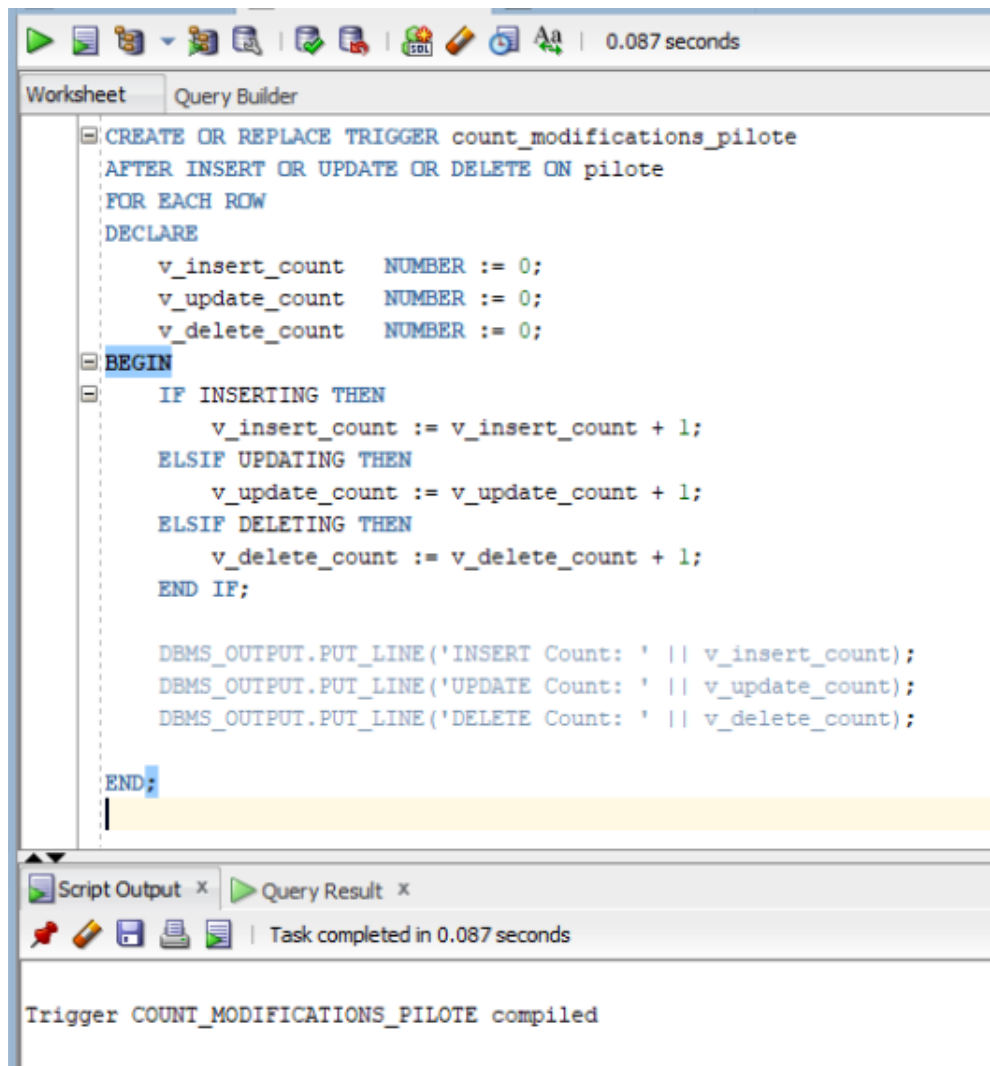
## 6. trigger verify\_nhvol :



The screenshot shows the SQL Developer interface with a query window titled 'Query Builder'. The query is for creating a trigger named 'verify\_nhvol'. The trigger is set to fire 'AFTER INSERT OR UPDATE ON AVION' and is configured to fire 'FOR EACH ROW'. The trigger body contains a 'DECLARE' section with a table type 'avion\_table' of type 'AVION.NUMAVION%TYPE', a variable 'v\_avion\_list' of type 'avion\_table', and a variable 'v\_avg\_hours' of type 'NUMBER'. The trigger body then contains a 'BEGIN' section with a 'SELECT' statement that calculates the average flight hours from the 'AVION' table. If the average is greater than 200,000, it raises an application error with the message 'La moyenne des heures de vol dépasse 200 000 heures.'.

```
CREATE OR REPLACE TRIGGER verify_nhvol
AFTER INSERT OR UPDATE ON AVION
FOR EACH ROW
DECLARE
    TYPE avion_table IS TABLE OF AVION.NUMAVION%TYPE;
    v_avion_list avion_table;
    v_avg_hours NUMBER;
BEGIN
    v_avion_list := avion_table(:NEW.NUMAVION);
    SELECT AVG(NBHVOL) INTO v_avg_hours
    FROM AVION;
    IF v_avg_hours > 200000 THEN
        RAISE_APPLICATION_ERROR(-20001, 'La moyenne des heures de vol dépasse 200 000 heures.');
```

## 7. comptabilisation du nombre de lignes modifiées :



The screenshot displays the Oracle SQL Developer interface. The top toolbar includes icons for running, saving, and other database operations, along with a timer showing 0.087 seconds. The main window is titled 'Query Builder' and contains the following SQL script:

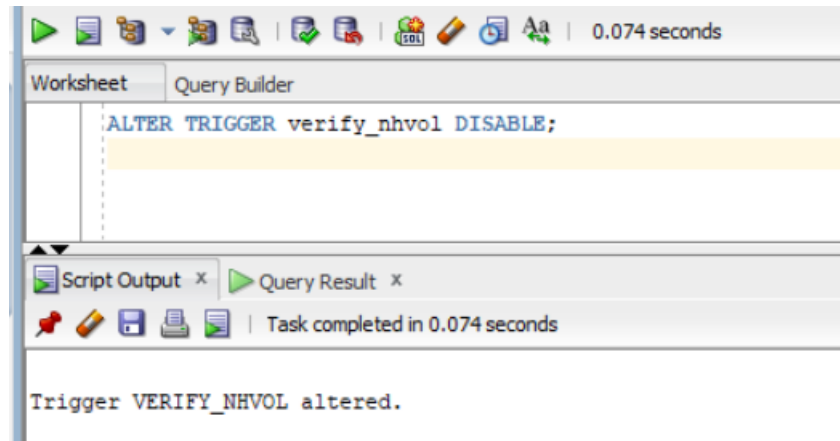
```
CREATE OR REPLACE TRIGGER count_modifications_pilote
AFTER INSERT OR UPDATE OR DELETE ON pilote
FOR EACH ROW
DECLARE
    v_insert_count    NUMBER := 0;
    v_update_count    NUMBER := 0;
    v_delete_count    NUMBER := 0;
BEGIN
    IF INSERTING THEN
        v_insert_count := v_insert_count + 1;
    ELSIF UPDATING THEN
        v_update_count := v_update_count + 1;
    ELSIF DELETING THEN
        v_delete_count := v_delete_count + 1;
    END IF;

    DBMS_OUTPUT.PUT_LINE('INSERT Count: ' || v_insert_count);
    DBMS_OUTPUT.PUT_LINE('UPDATE Count: ' || v_update_count);
    DBMS_OUTPUT.PUT_LINE('DELETE Count: ' || v_delete_count);
END;
```

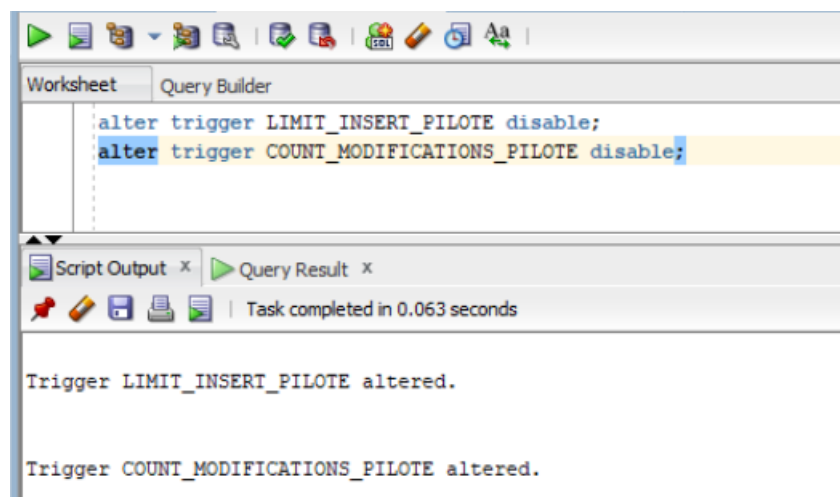
Below the script editor, there are two tabs: 'Script Output' and 'Query Result'. The 'Script Output' tab is active and shows the message: 'Task completed in 0.087 seconds'. Below this, the text 'Trigger COUNT\_MODIFICATIONS\_PILOTE compiled' is displayed.

## 8. Désactivation des triggers :

- Désactivation de trigger verify\_nhvol.



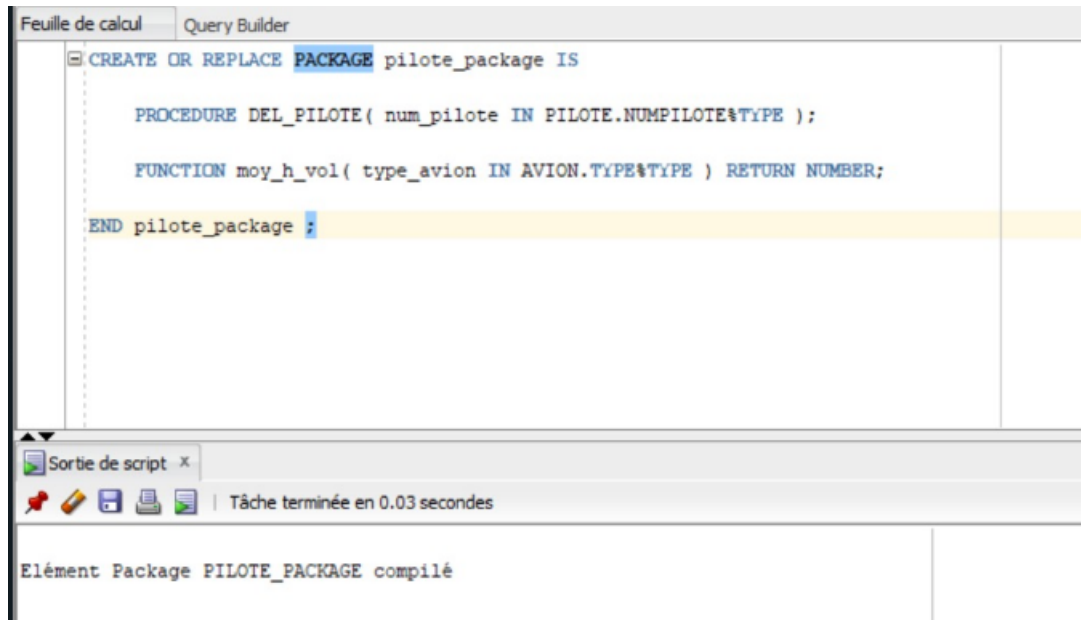
- Désactivation de tous les triggers de la table PILOTE.





## Packages

9. package pour suppression d'un pilote et calcul moyen des heures de vol :



The screenshot displays a SQL Query Builder window with a tab labeled 'Feuille de calcul' and 'Query Builder'. The main text area contains the following SQL code:

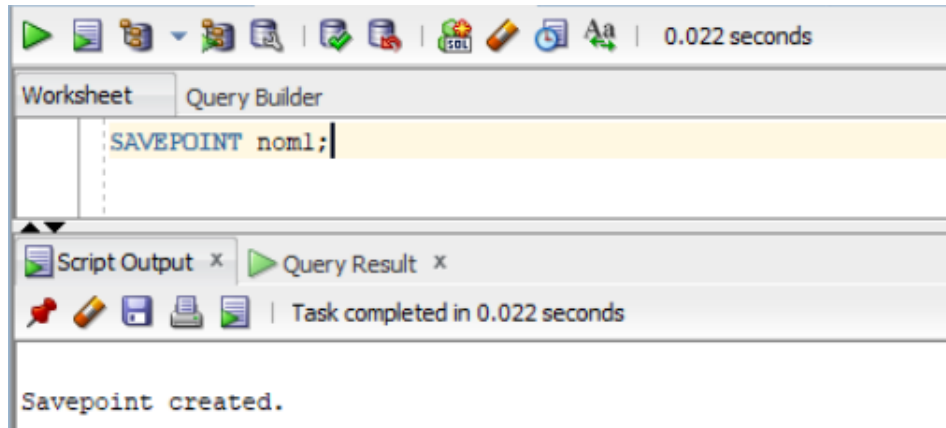
```
CREATE OR REPLACE PACKAGE pilote_package IS  
  
    PROCEDURE DEL_PILOTE( num_pilote IN PILOTE.NUMPILOTE%TYPE );  
  
    FUNCTION moy_h_vol( type_avion IN AVION.TYPE%TYPE ) RETURN NUMBER;  
  
END pilote_package ;
```

Below the code editor, there is a status bar with a 'Sortie de script' button and a message: 'Tâche terminée en 0.03 secondes'. At the bottom, a message box states: 'Elément Package PILOTE\_PACKAGE compilé'.

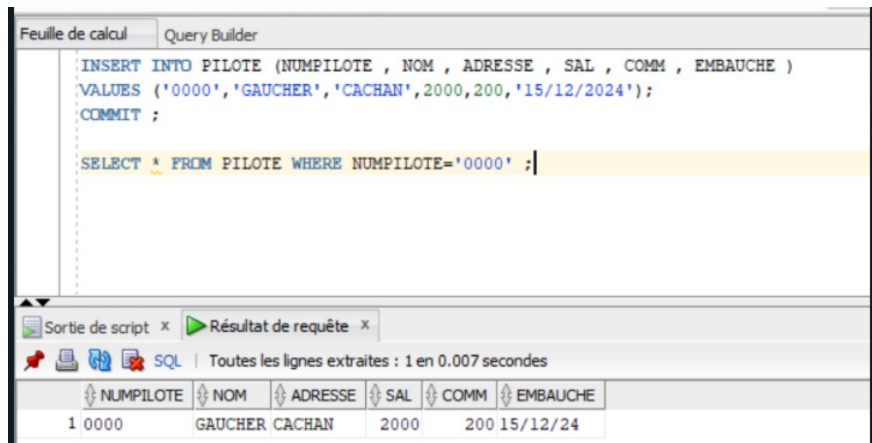
# Transactions

## 10. Visualisation de l'effet d'une transaction :

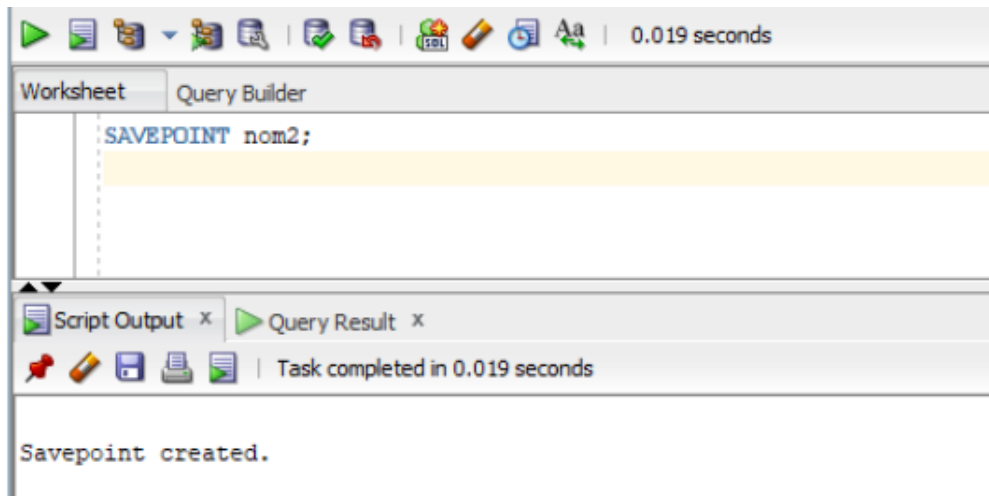
- Création d'un point de sauvegarde : savepoint "nom1";



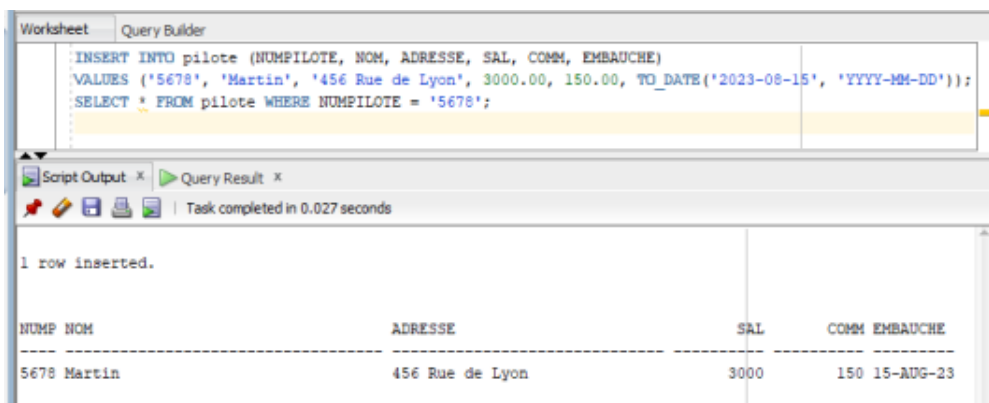
- Insertion d'un pilote, validation de l'insertion. et vérification de son existence.



- Création d'un point de sauvegarde : savepoint "nom2";



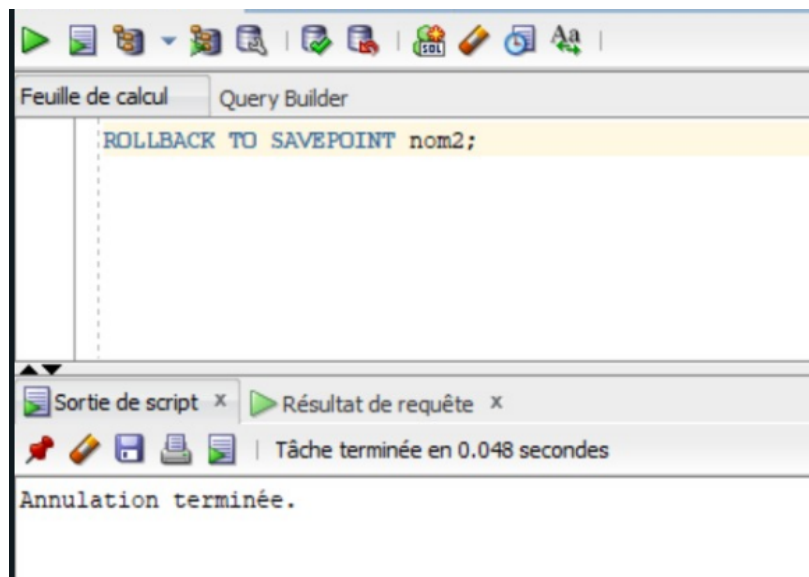
- Insertion d'un autre pilote, et vérification de son existence.



- Retour au savepoint "nom1";



- Retour au savepoint "nom2";



- Sélectionne tous les tuples de la table pilote.

The screenshot shows a database application window with a 'Query Builder' tab. The SQL query entered is 'SELECT \* FROM PILOTE'. Below the query, there are tabs for 'Sortie de script' and 'Résultat de requête'. The 'Résultat de requête' tab is active, displaying a table with 5 rows and 6 columns: NUMPILOTE, NOM, ADRESSE, SAL, COMM, and EMBAUCHE. The data is as follows:

	NUMPILOTE	NOM	ADRESSE	SAL	COMM	EMBAUCHE
1	0000	GAUCHER	CACHAN	2000	200	15/12/24
2	3452	ANDRE	NICE	4250	(null)	12/12/00
3	3421	BERGER	REIMS	3200	(null)	28/12/00
4	6723	MARTIN	ORSAY	4200	(null)	15/05/00
5	8843	GAUCHER	CACHAN	3875	(null)	20/11/00

- Explication du résultat obtenu.

Un commit valide définitivement la transaction en cours et supprime tous les save-points précédents. C'est pourquoi le rollback vers "nom1" échoue : le point de sauvegarde n'existe plus après le commit.